

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 02-010467

(43)Date of publication of application : 16.01.1990

(51)Int.Cl.

G06F 15/347
G06F 12/00

(21)Application number : 01-049762

(71)Applicant : ARDENT COMPUTER CORP

(22)Date of filing : 01.03.1989

(72)Inventor : GLEN S MILANCAR
JOHNSON STEVE

(30)Priority

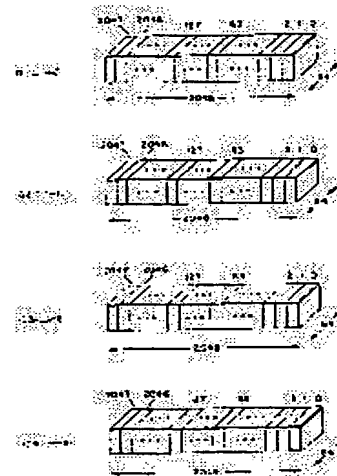
Priority number : 88 162738 Priority date : 01.03.1988 Priority country : US

(54) VECTOR REGISTER FILE

(57)Abstract:

PURPOSE: To improve processing speed while a signal port memory cell is used by constituting a memory circuit by means of a first bank and a second bank which can be cycled independent of the first bank.

CONSTITUTION: Four memory banks 101-104 constituted by static random access memories(SRAM) are used for a vector register file. The memory banks 101-104 are arranged when they can independently be cycled. SRAM having access time twice as much as the clock speed of the system is used for the memory banks 101-104, and the respective banks 101-104 can effectively be accessed twice in a prescribed system clock cycle.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

⑫ 公開特許公報(A) 平2-10467

⑤ Int. Cl.⁵

識別記号

庁内整理番号

⑬ 公開 平成2年(1990)1月16日

G 06 F 15/347
12/00D
3 0 4 M7050-5B
8841-5B

審査請求 未請求 請求項の数 6 (全9頁)

⑭ 発明の名称 ベクトル・レジスタ・ファイル

⑯ 特 願 平1-49762

⑰ 出 願 平1(1989)3月1日

優先権主張 ⑱ 1988年3月1日 ⑲ 米国(U S) ⑳ 162738

㉑ 発 明 者 グレン・エス・ミラン アメリカ合衆国 94118 カリフォルニア州・サン フラ
カー ンシスコ・セカンド アヴェニュー・120㉒ 発 明 者 ステイーブ・ジョンソ アメリカ合衆国 94301 カリフォルニア州・パロ アル
ン ト・メルヴィル アヴェニュー・144㉓ 出 願 人 アーデント・コンピュ アメリカ合衆国 94086 カリフォルニア州・サニイペー
ータ・コーポレーショ ル・ウエスト モード アヴェニュー・880
ン

㉔ 代 理 人 弁理士 山川 政樹 外3名

明細書の落し(内容に変更なし)

明 細 書

1. 発明の名称

ベクトル・レジスタ・ファイル

2. 特許請求の範囲

(1) メモリ回路の第1バンクと、

上記第1バンクとは独立して循環可能なメモリ
回路の第2バンクと、から成ることを特徴とするベクトル処理コンピ
ュータ・システムのベクトル・レジスタ・ファイ
ル。(2) サイクル時間nを有しているシステム・クロ
ックと、少なくとも約n/pのアクセス時間を有し、pが
少なくとも2である複数のメモリ回路と、から成ることを特徴とするベクトル・レジスタ・
ファイルを有するベクトル処理コンピュータ・シ
ステム。

(3) ベクトル・レジスタ・ファイルと、

上記ベクトル・レジスタ・ファイルをアドレス
するためオフセット値に加えられるシステム・ベース値を記憶するシステム・ベース・レジスタと、
から成ることを特徴とするベクトル処理コンピ
ュータ・システム。(4) ベクトル・レジスタ・ファイルと、システム・
ベース値を記憶するシステム・ベース・レジスタ
とを有するベクトル処理コンピュータ・システム
において、(a) 上記システム・ベース値にオフセット値を
加えて合計を出す過程と、(b) 上記合計において、上記ベクトル・レジス
タ・ファイルをアドレスする過程と、(c) 上記ベクトル処理コンピュータ・システム
がそのアクティブ・プロセスを変化する過程と、(d) 上記プロセスの変化に応じて、上記システ
ム・ベース・レジスタの値を変える過程と、から成ることを特徴とするベクトル・レジスタ・
ファイルをアドレスする方法。(5) アドレス可能なエレメントから成るベクトル・
レジスタ・ファイルを有するベクトル処理コンピ
ュータ・システムにおいて、

(a) ベクトル・レジスタの所定数を示す値 n を決定する過程と、

(b) 上記ベクトル・レジスタ・ファイルを、それぞれ p/n エLEMENTの深さを有する複数のベクトル・レジスタに論理的に分割する過程と、

から成ることを特徴とするベクトル・レジスタ・ファイルを複数のベクトル・レジスタに分割する方法。

(c) ベクトル・レジスタ・ファイルを有し、かつ上記ベクトル・レジスタ・ファイル内のアドレスから成る複数のオペランドを必要とするインストラクションを有するベクトル処理コンピュータ・システムにおいて、

上記各オペランドに関する上記アドレスの第1複数ビットを、ワードの第1半分に記憶する過程と、

上記各オペランドに関する上記アドレスの第2複数ビットを、ワードの第2半分に記憶する過程と、

から成ることを特徴とするオペランドを記憶す

る方法。

3. 発明の詳細な説明

〔発明の利用分野〕

本発明は、ベクトル処理コンピュータ・システムのベクトル・レジスタ・ファイルに関する。

〔発明の背景〕

ベクトル処理コンピュータ・システムにおいて、ベクトル・レジスタは、代表的には、計算のためベクトルを記憶するのに使用される。このようなシステムにおけるオペレーションは、同時にベクトルに対して実行される。たとえば、ベクトル処理システムにおける ADD VREG1, VREG2, VREG3 のようなオペレーションにより、ベクトル・レジスタ1における第1エレメントは、ベクトル・レジスタ2の第1エレメントに加えられ、その結果がベクトル・レジスタ3の第1エレメントに記憶される。同様に、ベクトル・レジスタ1の第2エレメントは、ベクトル・レジスタ2の第2エレメントに加えられ、その結果がベクトル・レジスタ3の第2エレメントに記憶される。ベクトル・

レジスタ1とベクトル・レジスタ2の各エレメントは、同じ様に処理され、ベクトル・レジスタ3の対応するエレメントに記憶される。

このようなコンピュータ・システムの処理速度を増すため、一般に、マルチポート・メモリを使用して、ベクトル・レジスタ・ファイルを構成している。このようなマルチポート・メモリは、単一のメモリ・サイクルにおいて、複数のメモリ・ロケーションに書き込みまたは読み出しができるという利点を有している。しかし、このようなマルチポート・メモリは、設計、構成、購入価格が単一ポートのメモリよりも高く、しかも単一ポートのメモリよりも容易には入手できず、また一般に単一ポートのメモリよりも処理速度が遅いという欠点を有している。

したがって、本発明の目的は、マルチポート・メモリに速度の利点を持たせながら、単一ポート・メモリ・セルを使用しているベクトル処理コンピュータ・システムのベクトル・レジスタ・ファイルを提供することである。

周知のベクトル処理システムは、一般にいくらか制限された大きさのベクトル・レジスタ・ファイルを使用している。たとえば、ミネソタ州ミネアポリス所在のクレイ・リサーチ社製の代表的なクレイ・コンピュータ・システムは、それぞれ、深さが64エレメントで、合計512エレメントの8つのベクトル・レジスタを有している。別の周知のシステムは、ベクトル・レジスタのスペースを2~4倍、使用している。通常、ベクトル・レジスタのスペースを増加するには、各ベクトル・レジスタの深さを(たとえば、128または256の深さに)増やすことによつて行なわれる。本発明の第2の目的は、ベクトル・レジスタのスペースの使用し得る量を増し、かつスペースの使用および管理のフレキシビリティを増すことである。

本発明の第3の目的は、ベクトルに対して作用するインストラクションを実行し、および任意のポイントにおいてベクトルのアクセスを必要とする計算ができるようなプログラミング・フレキシビリティを供給することである。このような計算

の例としては、復帰および累積計算またはその一方である。

〔発明の概要〕

本発明のベクトル・レジスタ・ファイルおよびベクトル・レジスタ・ファイルのアクセスを管理する方法について説明する。本発明は、複数のバンクで構成され、かつマルチポート・メモリのようにするため、システム・クロック速度の少なくとも2倍で動作するスタティック・ランダム・アクセス・メモリ(SRAM)を使用している。ベクトル・レジスタ・ファイルを構成するのにSRAMを使用することにより、多くの利点を得ることができる。

また、SRAMを使用することにより、コンピュータ・システムは、比較的多くのベクトル・レジスタ・ファイルのスペースを有するように構成できる。このスペースは、複数の文脈領域に分割でき、各文脈領域は、それぞれのプロセスを支持している。このような文脈領域を使用することにより、プロセスを切り換える場合、ベクトル・レ

ジスタ・ファイルの文脈情報をディスクにスワップする必要がない。また、本発明は、システム・ベース・レジスタ、ユーザ・ベース・レジスタ、および複数の文脈領域の使用を支持するシステム・リミット・レジスタの使用を開示している。

さらに、本発明は、ベクトル中の各エレメントすなわちセルを任意にアドレスをできるようにした、文脈領域をアドレスするアドレッシング方法を開示している。

以下、添付の図面に基いて、本発明の実施例に関し説明する。

〔実施例〕

本発明のベクトル・レジスタ・ファイルについて説明する。以下の説明において、ビット数、次元など様々な特定の記載は、本発明の理解を助けるためのものであつて、本発明はこれら特定の記載に限定されないことは、当業者には明白であらう。また、周知の技術については、本発明を不明瞭にしないよう詳細な記載は省略する。

ベクトル・レジスタ・ファイル

本発明において、ランダム・アクセス・メモリ(RAM)の複数のバンクとして、ベクトル処理コンピュータ・システムにベクトル・レジスタ・ファイルを配列している。ここで使用される目的に関して、語“バンク”は、メモリの独立的にサイクル可能な集合のことを意味している。すなわち、複数のバンクのそれぞれは、同じクロック・サイクルにおいて独立的にアドレスされることができる。

第1図は、本発明の実施例の詳細を示している。この実施例では、ベクトル・レジスタ・ファイルに対して、4つのメモリ・バンク101, 102, 103, 104が用いられている。各バンク101, 102, 103, 104は、1つのバンクが2,048個の64ビット・エレメントから成るように配列された8個の2,048×8スタティック・ランダム・アクセス・メモリ(SRAM)から成る。

本発明は、2つの技術を組合せることにより、8ポート・メモリと論理的に同じにしている。第1に、メモリを、4つのバンク、すなわち独立的

にサイクル可能な集合に配列して、4ポート・メモリと同様にしている(すなわち、バンク2, 3からの読み出しと同じクロック・サイクルにおいて、バンク0, 1に書き込みが行なわれる)。第2に、本発明は、システムのクロック速度の少なくとも2倍のアクセス時間を有するSRAMを使用して、所定のシステム・クロック・サイクルにおいて各バンクを有効に2回アクセスできるようにしている。本実施例では、システム・クロックは、120ナノ秒(ns)で進行しているので、多くの60nsのアクセス時間のSRAMを用いなければならない。35ns以上のアクセス時間のSRAMを使用することが望ましい。したがって、4つのバンクのそれぞれは、所定のどのクロック・サイクルでも2回アクセスされ、論理8ポートのメモリが得られる。

なお、これら方法は、マルチ・ポート・メモリを得るため、個々にまたは組合せて使用できることは、当業者には明白であらう。また、システム・クロック当り2回より多くのアクセスを行なうの

に、本発明の思想から離れることなく、異なるバンク数、異なる大きさのメモリ、または異なる速度のメモリを用いてもよいことは、当業者には明白であろう。

SRAMから成るベクトル・レジスタ・ファイルの本発明において使用することにより得られる利点は、SRAMが比較的安価で、複数の売手からSRAMを入手でき、しかも現在のSRAM技術から得られる速度が比較的速いことである。これら利点および他の要因により、周知のベクトル処理システムにおいて用いられている大きさよりも大きいベクトル・レジスタ・ファイルを使用できる。

本発明は、ベクトル・レジスタ・ファイルの管理および使用に関するいくつかの発明的技術および特徴を開示している。

本発明の特徴の1つとして、複数のバンクのそれぞれは、複数のベクトル・レジスタに分割される。たとえば、第2図に示すように、バンク201、202、203、204は、複数のベクトル・レジスタ210、211、212、213に分割される。第2図に

は、バンク・ナンバおよびオフセットを用いて示されている。たとえば、VR0(0)は、バンク0における第1ベクトル・レジスタで、VR0(32)は、ベクトル・レジスタが32エレメントの長さの、バンク0における第2ベクトル・レジスタで、VR0(64)は、バンク0における第3ベクトル・レジスタである。

1つの改変例では、ベクトル・レジスタは、バンク・ナンバとベクトル・ナンバを用いて示されている。たとえば、VR0,0は、バンク0、ナンバ0のベクトル・レジスタである。したがって、システムは、ベクトル・ナンバ(たとえば、0)をオフセットにトランスレートするよう応答可能である。VR0,1は、ベクトル0の第2ベクトル・レジスタである。そして、システムは、ベクトル・ナンバ(たとえば、1)をオフセットにトランスレートするよう応答可能である。たとえば、各ベクトル・レジスタが、32エレメントの幅である場合、システムは、ベクトル1をオフセット32にトランスレートする。

において、2,048エレメントのバンクのそれぞれは、64個の32エレメント・ベクトル・レジスタに分割される。

実際バンク301、302、303、304は、最大pまでのいずれかの数n個のベクトル・レジスタに分割される。なお、pは、1つのバンクにおけるエレメントの数である。各ベクトル・レジスタは、 p/n 個のエレメントを有している。pおよびnに与える特定の値は、個々のケースにおいて、ベクトル・レジスタが多くのエレメントを持つようにするか、あるいはベクトル・レジスタの数を多くするのかの長所短所のトレードオフにしたがい、通常のやり方に従って決めればよい。この通常のやり方は、ある場合にはハードウェア装置、またある場合にはソフトウェア的な手法に依存するものとなる。

本発明を使用することにより、ベクトル・レジスタは、1つのバンクのいずれかのエレメントにおいて開始する。エレメントをアドレスする特定の方法は、第6図において詳細に示されている。

本実施例において、特定のベクトル・レジスタ

第3の例としては、ベクトル・レジスタは、ベクトル・レジスタのナンバのみで示されている。したがって、システムは、ベクトル・ナンバをバンクおよびオフセットにトランスレートするよう応答可能である。たとえば、VR0は、バンク0、オフセット0にトランスレートされ、VR1は、バンク1、オフセット0にトランスレートされ、VR2は、バンク2、オフセット0にトランスレートされ、VR3は、バンク3、オフセット0にトランスレートされ、VR4は、バンク0、オフセット32にトランスレートされる。

これらの改変例は、互いにベクトル・ナンバがトランスレートされる速度と、プログラム・コードの携帯性とのトレードオフの関係にある。

ベクトル・レジスタの文脈領域

本発明では、前述したように、比較的多くのベクトル・レジスタ・スペースを使用することができ、また、どの単一プロセスにとつても、必ずしもベクトル・レジスタ・スペースの全部の量を用いる必要はない。したがって、本発明では、ベ

クトル・レジスタ・スペースを、複数のプロセス文脈領域に分割することもできる。システムは、プロセス間で切り換える際、ベクトル・レジスタ・ファイルからディスクのような他の蓄積媒体にプロセスの文脈データをスワップする必要なく、複数のプロセスを支持することができる。

第3図には、本発明の一実施例が示されており、ここでは、2,048要素のバンク301, 302, 303, 304は、各バンクにおいて、長さ256要素の8つの文脈領域310~317に分割されている。

第4図において、プロセスを進める場合、システムは、ベース・レジスタSBASE410を保持している。SBASE410は、バンクの最初からの（たとえば、要素0からの）現在の文脈領域のオフセットを含んでいる。たとえば、現在の文脈が、第3図の文脈領域312である場合、SBASE410は、値512を含んでいる。SBASE410を使用している場合、システムは、SBASE410の値を変化することにより、現在プロセスと、異

なる領域において文脈データを有するプロセスとの間で切り換えることができる。プロセスを進める時、SBASE410の値は、ベクトル・メモリの何らかの基準値となるオフセット値に加えられる。

SBASEレジスタは、さらに、ユーザのプロセスが、割り当てられた文脈領域より下ではオフセットをリファレンスしないですむようにもできる。さらに、SBASEレジスタを使用することによつて、プロセスを、それが割り当てられる特定の文脈領域に関係なく書き込むことができる。プロセスは、あたかも、それが常に第1文脈310において実行されるかのように進むべく書き込まれてもよい。

ベクトル・レジスタ・ファイルを、1とpの間の適当な数nの文脈領域に分割できることは、当業者には明白であろう。なお、pは、1つのバンクにおける要素の数である。文脈領域当りの要素の数は、 p/n である。特定の数nは、多くの要因により決まる。また、文脈領域の大きさを同じにしないで、比較的小さいプロセスに対

しては小さい方の文脈領域を使用し、比較的大きいプロセスに対しては大きい方の文脈領域を使用するように、文脈領域の大きさを変えることができることも、明らかである。

本発明は、さらに、現在の文脈領域の最終リファレンス可能要素を示すのに、第2レジスタSLIMIT411を使用している。SLIMIT411は、プロセスがその文脈領域におけるオフセットよりも大きいオフセットにおいて要素をリファレンスしないようにするため、使用される。ベクトル・レジスタ・ファイルにおける要素の何らかの基準値が、SLIMIT411に対して比較される。要素の基準値が、オフセットにおいてSLIMIT411よりも大きい場合、それは禁止され、エラー状態が示される。

本発明は、特定のプロセスがそれ自身の文脈スペースをさらに分割できるようにする第3レジスタ、UBASE412を使用している。代表的には、オペレーティング・システムは、SBASEおよびSLIMITレジスタを操作するよう応答可能であ

る。ユーザのプロセスはUBASEレジスタを操作する。たとえば、プロセスは、UBASE412の所定の値で固定オフセットに配置された各サブルーチンのための文脈データを有している。サブルーチンを実施する場合、ユーザは、UBASE412をUBASE412の所定の値にセットする。したがつて、サブルーチンは、UBASE412からオフセット値を供給することによつて、ベクトル・レジスタにおける要素をリファレンスすることができる。システムは、 $(SBASE410) + (UBASE412) + (所定のオフセット)$ を計算することによつて、ベクトル・レジスタ・ファイル内の実際のオフセットを計算する。

第5図は、文脈データとサブルーチンを共用する方法を示している。プロセスは、SBASE510とSLIMIT511の範囲内のいずれかにその文脈データを配置する。プロセスは、先ず、第1サブルーチンを実行する場合、そのUBASEレジスタをポイント518にセットして、領域530をリファレンスする。そして、プロセスは、UBASEレジ

スタをポイント521に変えて、領域531をリファレンスする。領域540は、どちらかのサブルーチンによりリファレンスされる共用変数または他の共用文脈情報を含んでいる。

UBASEレジスタの第2の使用目的は、限られた長さのアドレス・フィールドを用いている全システム・ベース領域(SBASEとSLIMITにより境界付けられた領域)をアドレスできるようにすることである。たとえば、システムのベース領域の長さが512エレメントで、かつ8ビット・アドレスがこの領域においてアドレスするのに使用されると仮定する。0から255のSBASEからの相対オフセットでエレメントをアドレスするため、UBASEは、0にセットされる。256から511のSBASEからの相対オフセットでエレメントをアドレスするため、UBASEは、256にセットされ、エレメントは(0から255)+UBASE(すなわち、256から511)としてアドレスされる。

第6図は、本実施例のインストラクションのオペランド・ディスクリプタ800を示している。本

フセットに加えられる。

ベクトル・レジスタ・アドレスの分割により、32ビット定数の代わりに、16ビット定数が、ベクトル・レジスタをリファレンスすることができる。なお、特定のアドレス・ビット構成は、本発明の思想の範囲において変更できることは、当業者には明白であろう。

また、本実施例において、16ビット以下の即値定数は、インストラクションの一部として記憶される。したがって、オペランド情報を記憶するのに、ワードの高次の半分だけを使用することによつて、本発明は、インストラクションのオペランド情報を記憶することができる。この方法は、インストラクションから分離したオペランド情報を検索しなくてもよいため、処理時間が短縮できる。

ベクトル・レジスタのセルを直接的にアドレスするか、またはスカラ変数を記憶するのにベクトル・レジスタを使用する場合、オペランド・ワード811の低次の半分が、オフセットのビット4〜

実施例のインストラクションは3つのオペランド、すなわちベクトル・レジスタ x (VR $_x$)、601(a)、601(b)、ベクトル・レジスタ y (VR $_y$)、602(a)、602(b)、ベクトル・レジスタ z (VR $_z$)、603(a)、603(b)から成る。

ベクトル・レジスタは、ベクトル・レジスタ・オペランドのビット9、8にバンク・ナンバーおよびビット7-0にオフセット値を与えることによりアドレスされる。

本実施例において、各ベクトル・レジスタ・オペランドのビット9-5は、ワード610の高次の半分に配置される。アドレスのビット9-5を使用して、本実施例の4つのバンクのいずれかが、ビット9、8でアドレスされ、バンク内のオフセットは、32の倍数でアドレスされる。たとえば、ベクトル・レジスタ・オペランドのビット9-5における00001₂の値は、バンク0、オフセット32をリファレンスする。11010₂の値は、バンク3、オフセット64をリファレンスする。前述したように、SBASEおよびUBASEの値は、オ

0にオフセット・アドレス情報の残りを供給する。また、第5図に関して述べたようにサブルーチンのためのデータをオーバーラップするには、オフセットのビット4〜0を使用する必要がある。

セル・レベルのアドレサビリティにより、コンボリユーションおよび復帰計算を実施することができる。たとえば、セル・レベルのアドレサビリティ特徴を使用して、次のようなコンボリユーション計算が行なわれる。

```
DO I=1, 32
  Y(I)=W(1)*X[I+1]
        +W(2)*X[I+2]
        +W(3)*X[I+3]
END
```

さらに、本発明の実施例は、ベクトルの各エレメントに対するオペレーションを含む全てのオペレーションを、シーケンシャル実行と一致したオーダで実行する。セル・レベル・アドレサビリティとの組合せにより、フィボナッチ・シーケンスのような復帰計算を直接的に計算できる。

このように、本発明は、ベクトル・レジスタ・ファイルおよびベクトル・レジスタ・ファイルのアクセスを管理する方法を提供する。

4. 図面の簡単な説明

第1図は、本発明において使用されるメモリ・バンクのブロック図、第2図は、メモリ・バンクを複数のベクトル・レジスタに分割する本発明による方法のブロック図、第3図は、メモリ・バンクを複数の文脈領域に分割する本発明の方法のブロック図、第4図は、本発明による、システム・ベース、ユーザ・ベース、およびシステム・リミット・レジスタを使用しているブロック図、第5図は、本発明による、システム・ベース、ユーザ・ベース、およびシステム・リミット・レジスタの使用、およびサブルーチン間での共有変数の使用を示したブロック図、第6図は、本発明において使用されるオペランド・ディスクリプタのブロック図である。

101, 102, 103, 104, 201, 202, 203, 204, 301, 302, 303, 304 バンク、

310~317 文脈領域。

特許出願人 アーデント・コンピュータ・コーポレーション

代 理 人 山 川 政 樹

図面の浄書(内容に変更なし)

FIG 1

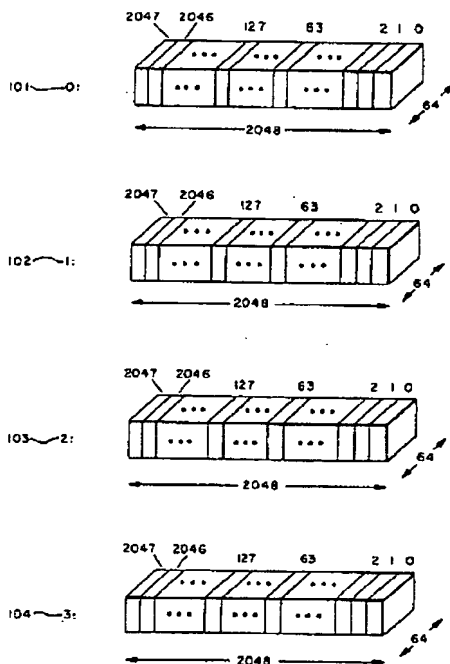


FIG 2

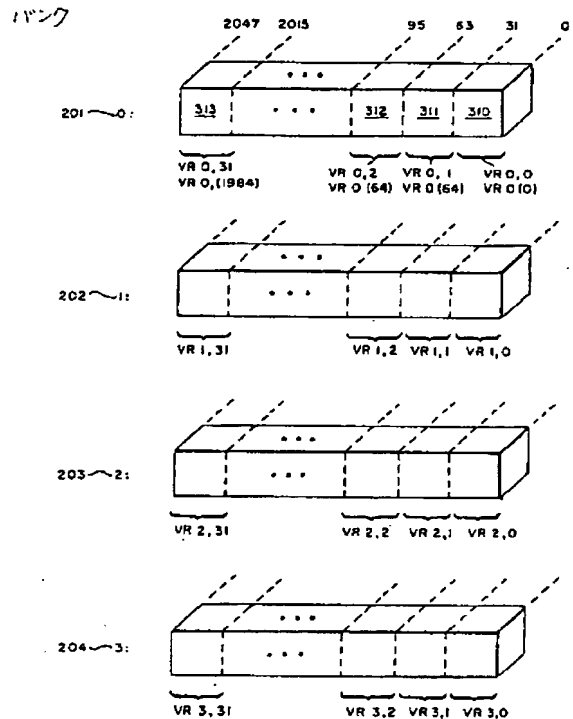
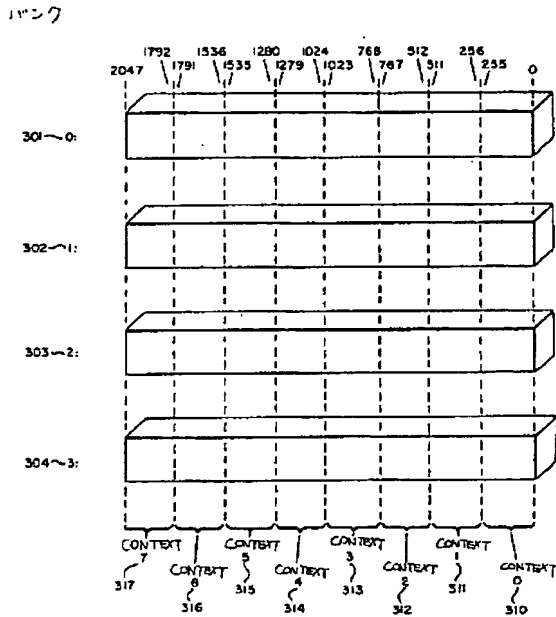
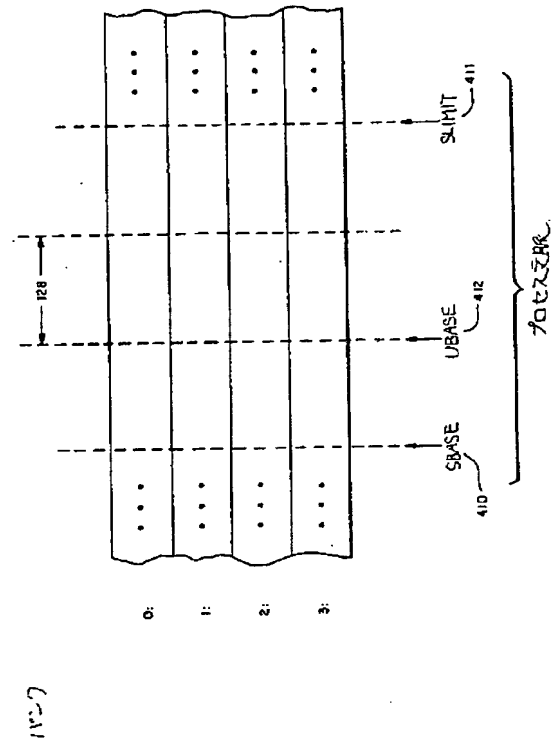


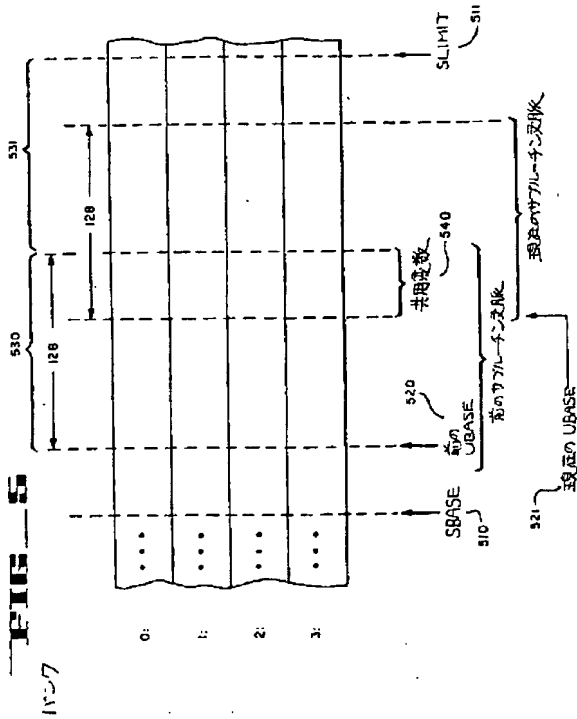
FIG. 3



FILE-4



5-111



FILE

[illegible]